Miniconf, 2016-12-19

Graph Recommenders
(in the Browser)

Bernd Malle, PhD Student
Benedikt Gansinger, BSc Student

b.malle@hci-kdd.org
b.gansinger@hci-kdd.org

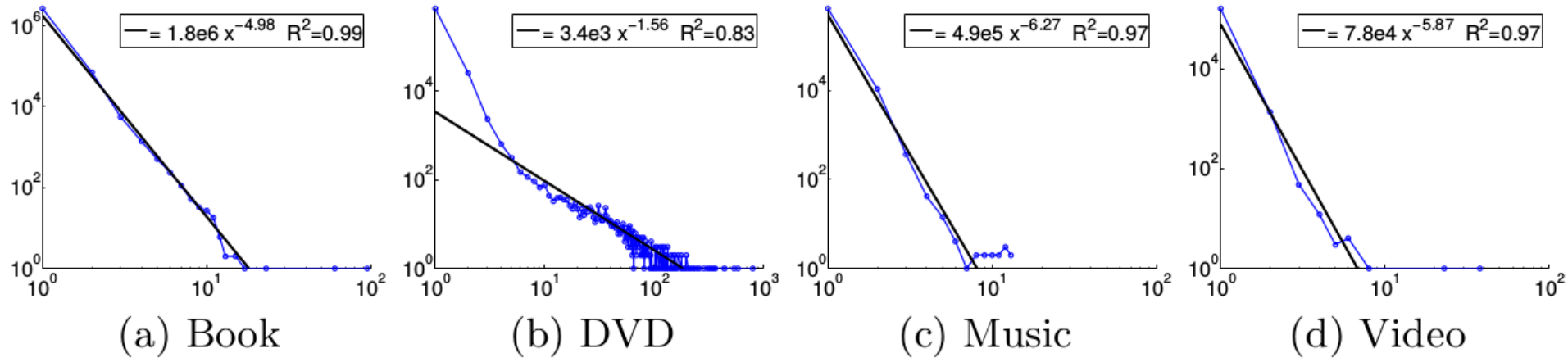
Introduction / TOC

1. Main idea for local graph recommenders
2. Desired outcome
3. Experimental Workflow
4. Visualization
5. Graph recommender algorithm
6. Datasets (anonymized FB graphs)
7. Enriching anonymized graphs
8. Modeling a stochastic process
9. Future work

Motivation: The local sphere (idea) 1/2

A study of recommendation cascades (clusters) in an online shopping network of millions of users showed that:

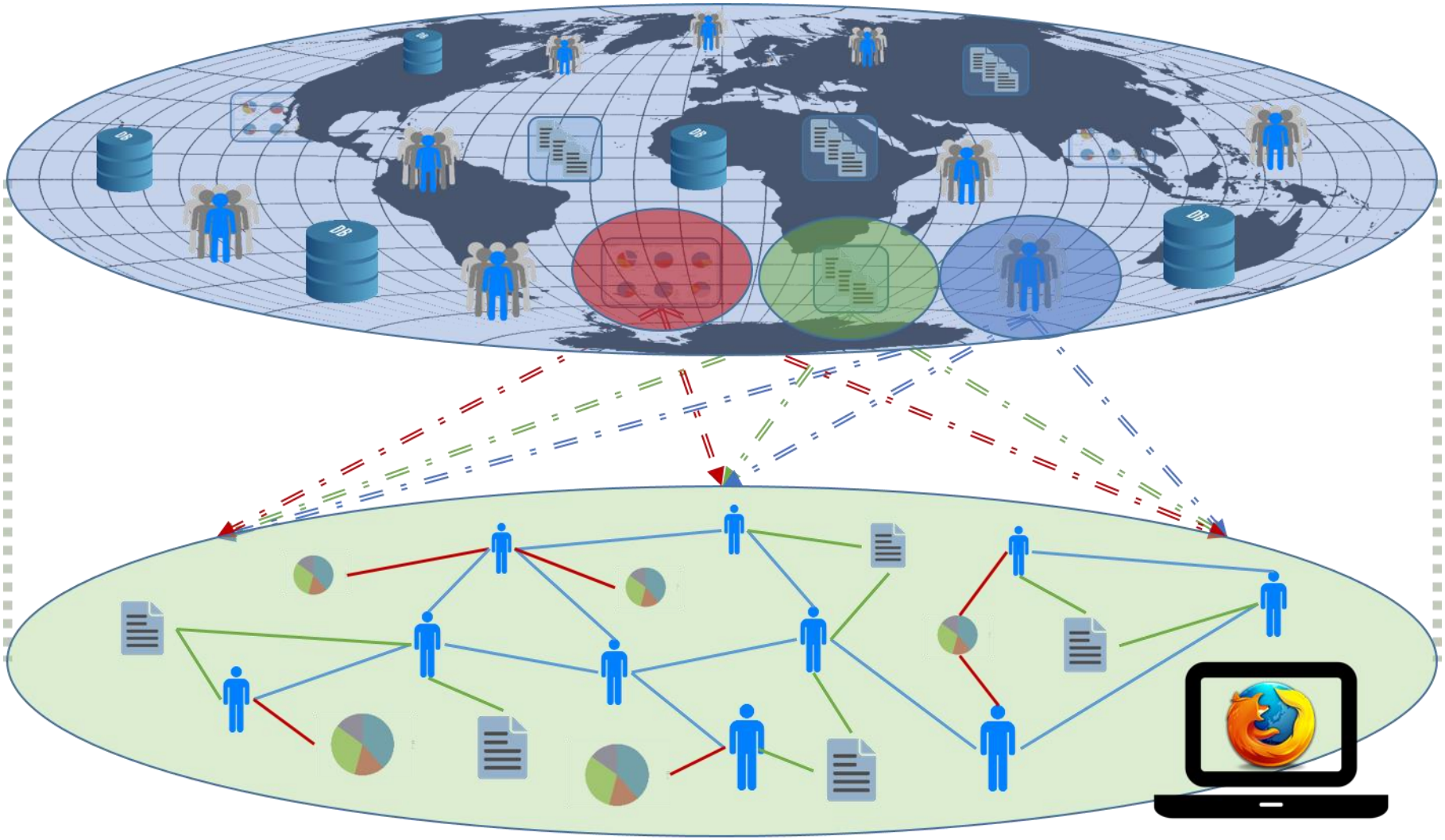
- The average size of cascades was relatively small (maximum ~10 f or all but DVDs)
- The radius of those cascades (graphs) is on the average less than 2 (degrees of separation)



Leskovec, Jure, Ajit Singh, and Jon Kleinberg. "Patterns of influence in a recommendation network." *Pacific-Asia Conference on Knowledge Discovery and Data Mining*. Springer Berlin Heidelberg, 2006.

Motivation: The local sphere (idea) 2/2

Pub/Sub in modern WebApps



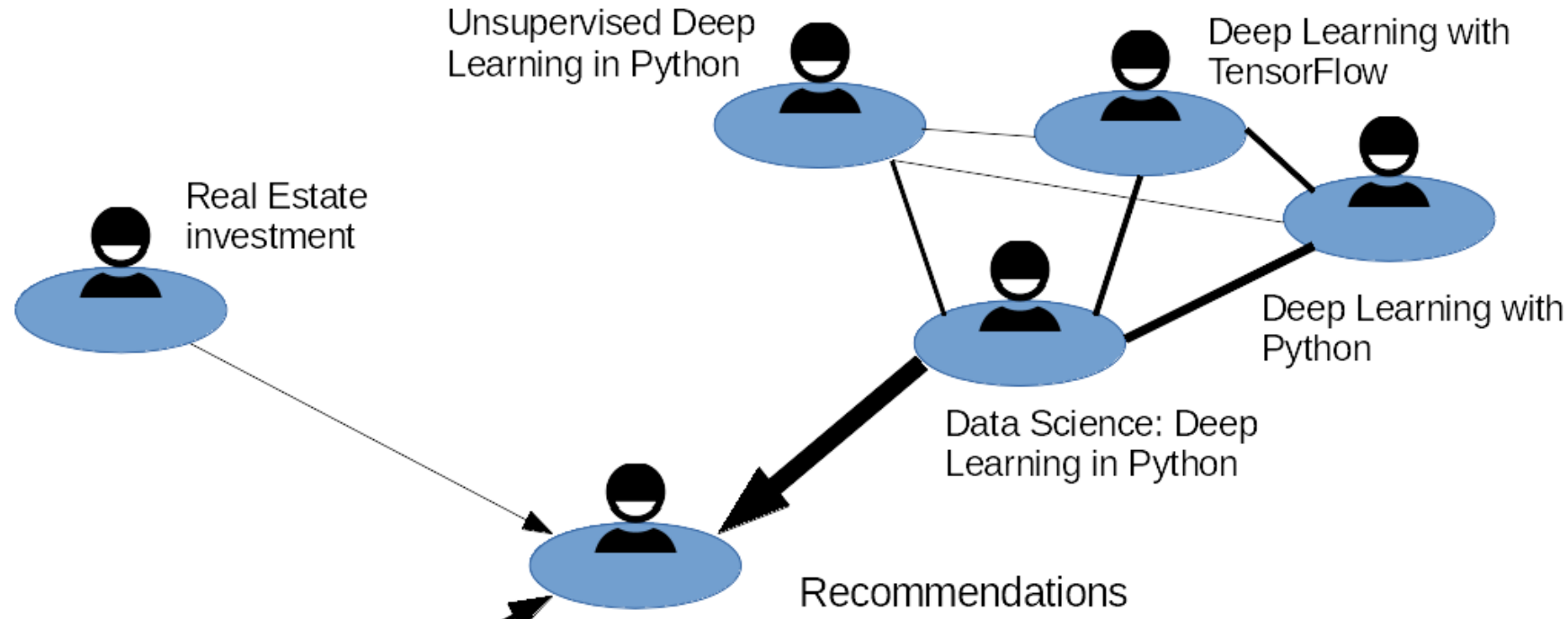
Desired outcome

- Better understanding of the influence factors in graph based recommendations
- Better understanding of the complexity / runtime behavior of graph based recommenders
  - Especially useful for larger graphs outside our experimental scope
- Basis for further experiments in ML on perturbed knowledge bases => “Playground project”
- Working software ;)

Experimental workflow

1. Researcher will navigate to the Website
2. Chooses from several pre-defined graphs
3. Recommendation mode: single / all users
   - We are recommending Udemy courses
4. Recommendations are *approved /rejected* or *rated*
5. The feedback alters the underlying feature vectors
   - This can also affect graph features / metrics (e.g. edge weights through user similarities)
6. Recommendations are re-computed… this cycle continues until researcher has enough data to write a paper about it ;)

Visualization



1. Data Science: Deep Learning in Python
2. Deep Learning with Python
3. Usupervised Deep Learning in Python
4. Deep Learning Prerequisites: Linear Regression in Python
5. Complete Python Bootcamp: Go from zero to hero in Python
6. Investment Portfolio Analysis with Python

.
.
.

Graph recommender algorithm

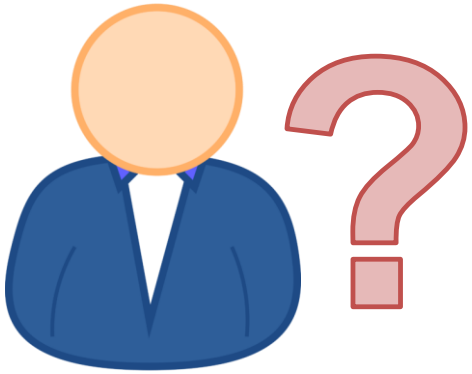
- Not yet determined, but recommenders can build on several factors:
  - Collaborative filtering – recommend what others have done / chosen (the typical amazon recommender)
  - Content based filtering (building a user profile)
    - A model of the user's preference
    - A history of the user's interaction with the recommender
  - Network structure

So which dataset is accessible to us ?

Datasets (anonymized FB graphs)

- Anonymized FB graphs provided by SNAP (Stanford Network Analysis Project)
- 10 FB EGO-graphs: social network from the perspective of one user => shows only relations among friends
- Just the structure is provided
- Content has to be re-generated

  (if necessary)



Enriching anonymized graphs 1/2

```
70 education;concentration;id;anonymized feature 426
71 education;concentration;id;anonymized feature 427
72 education;concentration;id;anonymized feature 428
73 education;concentration;id;anonymized feature 429
74 education;concentration;id;anonymized feature 430
75 education;degree;id;anonymized feature 21
76 education;degree;id;anonymized feature 431
77 education;degree;id;anonymized feature 313
78 education;degree;id;anonymized feature 432
```

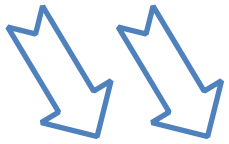
Random fake data generator…?

https://randomuser.me/api/portraits/women/68.jpg

```
{ first_name: 'Monique',
  middle_name: 'Imogene',
  last_name: 'Weimann',
  gender: 'female',
  avatar_url: 'https://randomuser.me/api/portraits/women/68.jpg' }
{ work_end_date: 2015-04-24T08:53:04.456Z,
  work_start_date: 2011-06-11T08:07:46.529Z,
  graduation_date: 2011-03-04T10:59:40.290Z,
  birth_date: 1991-03-09T10:59:40.000Z }
{ employer_id: 'Adams - D\'Amore',
  from_id: null,
  with_id: null,
  location: '00588 South Treside',
  position: 'District Web Administrator',
  projects: [ 'Refined Fresh Chair' ] }
{ hometown: '37317 South Charlotteside',
  location: '26154 South Athena',
  languages: [ 'English', 'Mandarin' ],
  locale: 'en_US' }
```

Enriching anonymized graphs 2/2


```
Course #  Course Title  Level
16.00 Introduction to Aerospace Engineering and Design  Undergraduate
16.00AJ Exploring Sea, Space, & Earth: Fundamentals of Engineering Design
16.01 Unified Engineering I, II, III, & IV (Fall 2005)  Undergraduate
16.02 Unified Engineering I, II, III, & IV (Fall 2005)  Undergraduate
16.03 Unified Engineering I, II, III, & IV (Fall 2005)  Undergraduate
16.04 Unified Engineering I, II, III, & IV (Fall 2005)  Undergraduate
16.050  Thermal Energy  Undergraduate
16.06 Principles of Automatic Control Undergraduate
```



```
Config.UNIVERSITIES holds 7626 distinct entries.
Config.DEGREES holds 195 distinct entries.
Config.SCHOOL_TYPES holds 3 distinct entries.
Config.CONCENTRATIONS holds 152 distinct entries.
Config.CLASSES holds 37 distinct entries.
```

OK – so we have enough data, but how to assign them realistically to our users considering the FB graph?

Modeling a stochastic process

- Our anonymized graph might give us different schools (anonymous features) for 2 connected users

- But which schools should we assign? Random ones?
  - Certainly, the likelihood for 2 people to be acquainted is higher if they attend geographically close schools (MIT / Harvard) than far apart (MIT / TU Graz)
  - What about qualifications / interests / languages etc.??

- We need to base our data generators on stochastic processes resembling reality => Probably a significant project in itself !

Future work

1. Scaling of graph algorithm to real-sized DBs?
2. Implementation of local sphere idea?
   - How to connect to a super-graph
3. Perturbation of graphs:
   - adding / removing nodes / edges
   - Perturbing node feature vectors
   - Targeting specific graph metrics:
     - Centralities
     - Components
     - Flow properties (throughput etc. …)

Thank you!